Cloud Preparation - Mirror STAR GitLab to Cloud GitLab

1. Visit http://gitlab.star.nesdis.local and log in;
2. Make a project in Cloud GitLab with the same project name as in the STAR GitLab;
3. In the project page in STAR GIT, go to Settings -> Repository -> Mirroring Repository -> Expand and enter the following at "GIT repository URL": http://user.lastname@star-gitlab-extIernal-ef5c5e00a8959c1f.elb.us-east-1.amazonaws.com/user.lastname/project-name.git
4. Please note that you can only push changes from STAR side. If you push changes from cloud side to cloud GitLab, that repository will become protected and you will not be able to mirror changes from STAR side anymore. In short, only use cloud GitLab to check out repositories, do not push anything to it.

Libraries Needed for Linux 7 Environment

1. java
2. wget
3. eog
4. xmgrace
5. gnuplot
6. kdiff3
7. valgrind
8. nco-static
9. nco-devel
10. netcdf-static
11. netcdf-devel
12. netcdf4-python.x86_64
13. hdf5
14. gcc-c++.x86_64
15. gcc-gfortran.x86_64
16. make
17. patch
18. csh
19. zlib-devel.x86_64
20. libpng-devel.x86_64
21. bzip2-devel.x86_64
22. libXt-devel.x86_64
23. expat-devel.x86_64
24. bison.x86_64
25. flex.x86_64
26. openssl-devel.x86_64

27. bc.x86_64
28. unzip.x86_64
29. java-1.8.0-openjdk-devel.x86_64
30. imake.x86_64
31. tcl-devel.x86_64
32. flex-devel.x86_64
33. autoconf.noarch
34. automake.noarch
35. git.x86_64
36. ksh.x86_64
37. icu.x86_64
38. boost.x64_64
39. libxerces
40. perl-devel.x86_64
41. perl-YAML.noarch
42. perl-XML-Simple

MSL12 code preparation

1. Checkout from Cloud GitLab;
2. Cloud IT installed glibc-static;
3. Modify Makefile and add -lpthread to the LINKLIBS:
   LINKLIBS = -lsvml -limf -L./lib -lgsl -lgslcblas -loli -lanc -lnav -lgenutils -lmfhdf -ldf -ljpeg -lnetcdf -lhdf5_hl -lhdf5 -lgeotiff -ltiff -lproj -lz -lsz -lm -lpthread -lstdc++
4. Remove the MSL12 src/lib/libpthread.a or rename to libpthread.a.SAVE. This will force it to use the systems pthread lib in /usr/lib64;
5. Compile the MSL12 code

ADL version 5.3.16 Mx3 (revised to produce gain state file and run in Linux 7 environment)

1. Checkout from Cloud GitLab;
2. Modify the ADL/build/envSetup.ksh code to re-direct all the COTS location to /share/software/assistt/ADL/ADL53_BLK2_COTS/
3. Many COTS location environment variables (9 in this case) need to be individually revised which are listed as below:
   ADL_HOME, COTS_HOME, JAVA_COTS, COTS_PYTHON_HOME, JAVA_HOME, LOG4J_JAR_FILE, COTS_DIGESTER_HOME, COTS_SWINGX_HOME, COTS_SQLITE_HOME
4. Need to add one line after the compiler environment variable definitions:
   export USER_CCFLAGS="-fpermissive"
5. Run the buildAdl.csh script until it stops.

6. Go to the verified RDR source code folder and edit Makefile to enable binary executable generation.
7. Perform make in that folder.
8. Some database related libraries/binaries also need to be built.
9. The ADL building is not complete, but adequate to get the gain state files generated.